Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A method of aligning instructions in a processor comprising:

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area including a plurality of sub-buffers, each sub-buffer storing a unit instruction width, with an instruction of greater than a unit instruction width stored in more than one sub-buffer;

aligning a first instruction from said buffer areas;

decoding a size of the first instruction;

selecting at least one of said plurality of sub-buffers ~~areas~~ from which to output said first instruction on an output part;

during said outputting, ~~first determin~~ determining a beginning of a second instruction from selected ones of the plurality of sub-buffers based on the size of the first instruction, decoding the size of the second instruction, and ~~second~~ determining whether processing the second instruction will deplete said plurality of ~~buffers~~ buffer areas; and

based on said ~~second~~ determining whether processing the second instruction will deplete said plurality of buffer areas, instructing the plurality of ~~buffers~~ buffer areas to receive additional instructions.

2. (Cancelled)

3. (Currently amended) The method of Claim 1, further comprising comparing a most significant bit of a pointer to a first ~~of the plurality of~~ sub-buffer[[s]] to a most significant

bit of a pointer to a second ~~of the plurality of~~ sub-buffer[[s]] to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

4. (Currently amended) The method of Claim 1, further comprising storing a first instruction across a plurality of ~~storage elements~~ sub-buffers prior to processing the instructions.

5. (Original) The method of Claim 1, further comprising adding the size of the first instruction to a current instruction position to determine the beginning of the second instruction.

6. (Original) The method of Claim 1, further comprising aligning ahead a number of cycles equal to a cache latency.

7. (Original) The method of Claim 1, further comprising aligning instructions in a digital signal processor.

8. (Currently amended) The method of Claim 1, further comprising issuing a request to a memory to reload the plurality of ~~buffers~~ buffer areas.

9-17. (Cancelled)

18. (Currently amended) The processor of Claim 27, wherein the transition detector compares a most significant bit of a pointer to a first ~~of the plurality of sub-buffers~~ subpart to a most significant bit of a pointer of a second ~~of the plurality of sub-buffers~~ subpart to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

19. (Previously presented) The processor of Claim 27, wherein the processor aligns ahead a number of cycles equal to a cache latency.

20. (Currently amended) The processor of Claim ~~21~~ 27, wherein the processor is a digital signal processor.

21. (Currently amended) An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to align instructions in a processor, the instructions causing the machine to:

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area including a plurality of sub-buffers, each sub-buffer storing a unit instruction width, with an instruction of greater than a unit instruction width stored in more than one sub-buffer;

decode a size of a first instruction from said plurality of buffer areas;

select at least one of said plurality of sub-buffers ~~areas~~ from which to output said first instruction on an output part;

during said outputting, ~~first~~ determining a beginning of a second instruction from selected ones of the plurality of sub-buffers based on the size of the first instruction, decoding the size of the second instruction, and determining whether processing the second instruction will deplete ~~said~~ the plurality of ~~buffers~~ buffer areas; and

based on said ~~second~~ determining whether processing the second instruction will deplete the plurality of buffer areas, instruct the plurality of ~~buffers~~ buffer areas to receive additional instructions.

22. (Cancelled)

23. (Currently amended) The apparatus of Claim 21, wherein a most significant bit of a pointer to a first ~~of the plurality of~~ sub-buffer[[s]] is compared to a most significant bit of pointer to a second ~~of the plurality of~~ sub-buffer[[s]] to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

24. (Currently amended) The apparatus of Claim 21, wherein a first instruction is stored across a plurality of ~~storage elements~~ sub-buffers prior to processing the instructions.

25. (Currently amended) A method of processing instructions within a processor, comprising:

storing instructions of different widths within a cache having a plurality of buffer areas, each buffer area having a plurality of subportions, each subportion in the cache storing a unit instruction width, where an instruction of unit width takes up a single subportion in the cache, and an instruction of more than said unit width takes up more than one ~~subportions~~ subportion within the cache;

multiplexing each of the subportions of said cache to an output point, and selecting at least one of said cache subportions as a current instruction;

during said selecting said current instruction, predicting which of said ~~instructions~~ buffer areas within said cache will be depleted of instruction data within a number of cycles approximately equal to a latency of the cache, and instructing loading of that number of ~~buffers~~ buffer areas with additional instruction information.

26. (Currently amended) A method as in claim 25, wherein said predicting comprises comparing a most significant bit of a pointer to a first subportion, to a most significant bit of a pointer to a second subportion, to determine if any of the ~~subportions~~ buffer areas will be depleted.

27. (Currently amended) A processor comprising:

a plurality of buffer areas, each buffer area adapted to store a plurality of instructions of different ~~width~~ widths in a plurality of subparts, each of said subparts storing a unit instruction width, and said instructions of greater than a unit instruction ~~widths~~ width being stored in multiple said subparts;

a multiplexer, connected to said plurality of subparts, and selecting and aligning at least one of said plurality of subparts from any of said subparts within said buffer areas as a current instruction; and

a predictor, operating to predict when at least one of the plurality of buffer areas will be empty, and to send a signal to instruct said at least one of the plurality of buffer areas to load another instruction data.